# System Integration

## What are microservices?



Patrick Stolc
M.Sc. Robotics Engineering

# Agenda: Microservices & Domains

- Discussion: What are Microservices?

- Presentation: The project you'll work on

- Workshop: Identify core domains

- Feedback: Inter-group feedback session

# Agenda: Microservices & Design

- Discussion: Microservice design principles

- Workshop: Define boundaries and service interactions

- Feedback: Inter-group feedback session

# Agenda: Wrap-up

- Workshop: Finalize your microservices design

- Presentation: 2 minute design presentation

# What are Microservices?

💾 **Definition and properties**

- A microservice is a small, independent, and loosely coupled service that can be developed, deployed, and scaled independently.

- Each microservice is designed to perform a specific function or task.

# What are Monoliths?

## 💾 Definition

- A monolithic architecture is a single, cohesive application that is built as a single unit.

- All components are tightly coupled and share the same codebase.

# Microservices vs Monoliths

💾 **Advantages of Microservices**

- **Scalability**: Microservices can be scaled independently.

- **Flexibility**: New microservices can be added or removed without affecting the entire application.

- **Resilience**: If one microservice fails, it doesn't bring down the entire system.

- **Ease of Development**: Each microservice can be developed by a different team, allowing for specialized expertise and faster development cycles.

# Microservices vs Monoliths

💾 **Disadvantages of Microservices**

- **Complexity**: Managing a large number of small services can be complex.

- **Overhead**: The overhead of managing and monitoring a large number of services can be significant.

- **Data Consistency**: Ensuring data consistency across services can be challenging.

# Microservices vs Monoliths

## 💾 Advantages of Monoliths

- **Simplicity**: A single, cohesive application is easier to understand and maintain.

- **Performance**: Monolithic applications can be more performant due to less overhead.

- **Ease of Development**: All components are in one place, making it easier to develop and test.

# Microservices vs Monoliths

💾 **Disadvantages of Monoliths**

- **Complexity**: As the application grows, it becomes harder to manage.

- **Scalability**: Scaling a monolithic application can be challenging.

- **Restrictions**: Not always possible to choose the most suitable technlogy.